Remarks

These Remarks are in reply to the Office Action mailed April 7, 2004. Claims 1-30, 47-50, 83-91were pending in the Application prior to the outstanding Office Action.

I. Status of Claims

Claims 1-30, 47-50 and 83-91 were pending in the Application prior to the outstanding Office Action. Claims 1, 47, 85 and 86 are currently being amended, leaving claims 1-30, 47-50 and 83-91 for Examination. For at least the reasons set forth below, Applicants respectfully request reconsideration of all the outstanding rejections and objections and allowance of the claims.

II. Amendment to Specification

The abstract has been amended to overcome the objection set forth in the Office Action.

III. Summary of Claim Rejections

In the most recent Office Action mailed April 7, 2004, the Examiner appears to have withdrawn the previous 35 U.S.C. §103(a) rejections that were based on U.S. Patent No. 5,617,567 (Doktor) in view of U.S. Patent No. 5,386,571 (Kurz). However, the Examiner is now asserting that the claims are obvious based on Doktor in view of U.S. Patent No. 5,504,879 (Eisenberg). More specifically, claims 1-30, 47-50 and 83-91 were rejected under 35 U.S.C. §103(a) as being unpatentable over U.S. Patent No. 5,617,567 issued to (hereinafter "Doktor") in view of U.S. Patent No. 5,504,879 issued to Eisenberg (hereinafter "Eisenberg").

{In reviewing the Office Action, Applicants' representative noticed that on page 2, in Section 3, the Examiner stated "However, Hancock discloses such limitations." There was, however, no patent number provided for Hancock, and no further discussion of Hancock. Accordingly, Applicants' representative called the Examiner and the Examiner confirmed that this was an error, and that the sentence should have stated "However, Eisenberg discloses such limitations."}

IV. Discussion of Claims

A. Claims 1 and 47

Claims 1 and 47, as amended, both claim the following features:

a) multiple operation records each storing data relating to one or more historical operation involving at least one entity, each said operation record comprising data recording the operation, and data defining a date associated with the operation;

each said entity being an identifiable thing within a business or other undertaking to which information resulting from a transaction, measurement or other such assignment can be related; and

b) each said entity being represented by a single corresponding entity record, said ~~multiple~~ entity records storing data indicating relationships between said entities, and each said relationship being associated with a historical period of validity.

These claims have been amended to define the invention more clearly. Support for this amendment can be found, for example, on: page 18, line 1 to page 19, line 13; page 28, lines 20 –21; page 31, lines 13 – 22; page 69, line 21 to page 70, line 2; claim 10.

In the invention as claimed in currently amended claims 1 and 47, each entity has only one (i.e., a single) corresponding entity record. When the circumstances involving that entity change, the database needs to be updated to reflect those changes. For example, if a product manager moves from one department to another.

In the invention, a new association record is created to identify these changes as shown on page 18, line 1 to page 19, line 13. A new entity record is not created. Only one entity record per entity is maintained during the lifetime of that entity. This allows the business structure to be maintained in the database in an easy and efficient manner and for queries to the data to be carried out efficiently at any point in time. Repetition of information in records is not required and so storage space is saved.

In contrast, the Eisenberg patent creates a new entity record every time there is a change, thus resulting in a huge amount of data that becomes increasingly hard to query and maintain (See cols 17 & 18 of Eisenberg).

Further, it is defined in claims 1 and 47 that entity records store data indicating relationships between entities. For example, a sales manager and an area of sales may be the two entities that have a relationship, i.e., sales manager A is responsible for sales area C. The relationship is also associated with a historical period of validity; for example, sales manager A was only responsible for sales area C during the year 2001. With this arrangement it is possible to update the business organisation stored in a database without having to go through the expensive and time-consuming task of reformatting database structures and producing new entity records every time a change occurs.

As claims 1 and 47 are novel and non-obvious over Doktor in view of Eisenberg, it is asserted that all claims dependent on claims 1 and 47 (i.e., claims 2-30, and claims 48-50) are also patentable.

B. Claim 83

In addition to the arguments raised in the section above in relation to claims 1 and 47, which are also pertinent to claim 83, the following arguments should also be taken into account in relation to claim 83.

Claim 83 claims:

A data processing system comprising a data storage device and a processor programmed to read data from, and write data to, said storage device, in which said storage device stores a time variant data model to which data in a data structure conforms, the data model generated by the processor and representing the relationships between a plurality of classes of entities, said storage device further storing:

a) multiple operation records each storing data relating to one or more historical operations involving at least one said entity conforming to one of said classes, each said operation record comprising data recording the operation, and data defining a date associated with the operation, each said entity being

an identifiable thing within a business or other undertaking to which information resulting from a transaction, measurement or other such assignment can be related; and

b) multiple entity records and association records which conform to the data model, each of the multiple entity records comprising an entity record for each said entity conforming to one of said classes, said association records storing data indicating past or present relationships between a pair of said entities, and each said entity record containing data associating each said relationship with a historical period of validity.

In the most recent Office Action of April 7, 2004, the Examiner has not raised any arguments as to why this claim is not valid in light of the cited prior art. The claims include the feature of a time variant data model and data structure stored alongside each other within the storage device. The data structure is a particular example of the data model. The data model uses metadata to describe the classes of entities that make up the data model. Each of the entities stored in the data structure conform to one of the classes of entities within the data model. Figure 7 of the present application shows an example of the data model, and Figures 8a and 8b show examples of the data structure.

Neither Doktor nor Eisenberg, alone or in combination, indicate that a data model and a data structure can be stored alongside each other, and so do not have the advantage of being able to modify the data model and the data structure in order to keep, for example, a business organisation database up to date with minimum cost and effort. Indeed, neither Doktor nor Eisenberg discuss metadata (i.e., that which forms the data model) in any form. Doktor merely discloses storing entities and relationships. Eisenberg merely discloses storing a 'part', see col 10, line16, which is either an entity or relationship. Neither discloses storing classes of entities to form a data model, let alone a time-variant data model stored alongside a time variant data structure.

For at least the reasons set forth above, it is therefore asserted that claim 83 is novel and unobvious over Eiesenberg and Doktor, alone or in combination.

C. **Claims 48 and 84**

Claim 48 claims:

A data processing system comprising a data storage device and a processor programmed to read data from, and write data to, said storage device, in which said storage device stores multiple operation records each storing data relating to one or more historical operation involving at least one entity,

each said entity being an identifiable thing within a business or other undertaking to which information resulting from a transaction, measurement or other such assignment can be related;

and multiple entity records storing data indicating relationships between said entities,

wherein the entity records comprise a hierarchical structure, in which at least a first entity record relates to a specific entity, and a second to a more generic entity encompassing said specific entity, said entity records including link data linking said first and second entity records whereby to allow said processor to traverse said hierarchy,

said processor being arranged to generate output data by inputting instructions defining one or more selected entity dimensions across which said output data is to be distributed.

It is admitted in the Office Action that Doktor does not explicitly disclose "wherein the entity records comprise a hierarchical structure, in which at least a first entity record relates to a specific entity, and a second to a more generic entity encompassing said specific entity, said entity records including link data linking said first and second entity records whereby to allow said processor to traverse said hierarchy." However, the Examiner then asserted that Eisenberg discloses this feature at col. 3, lines 20 – 32 and col. 24, lines 39 – 48. Applicants respectfully disagree. Eisenberg is merely discussing at column 3 problems associated with attempting to comply with the principle of "relationship preservation during update" as well as complying with the principle of "relationship absence after add." It discusses a parent - child hierarchy of two different versions (or variants) of data stored in a database, i.e., Production and Test. For example, a test software version and a production software version wherein the production software is produced after the test software has been approved. Entity records A and B are described in relation to the two different versions of data. The terms Production and Test merely relate to the different

versions of the software, for example, and not to the entity records, A and B. The entity records A and B themselves do not relate to one specific and one generic entity, but merely correspond to two different entities.

Further, at col. 24, lines 39 – 48, Eisenberg merely discloses different versions (variants) of a data structure formed in a number of different hierarchical structures. Only one of these hierarchical structures is nominated as the primary version and is the only version that can be used to perform updates. This is so that, in the example of software versions, only the latest up-to-date version is amended in order to avoid any errors. It does not discuss entity records formed in a hierarchical structure. Further, it does not disclose a first entity record relating to a specific entity, and a second to a more generic entity encompassing said specific entity.

There is no disclosure in either Doktor or Eisenberg of a hierarchical structure of entity records, in which at least a first entity record relates to a specific entity, and a second entity to a more generic entity encompassing said specific entity. For at least this reason, Applicants respectfully request that the 35 U.S.C. 103 rejection of claim 48 be withdrawn.

Further, as claims 49 and 50 are dependent on claim 48, they are also novel and unobvious for at least the reasons given above.

The arguments raised in relation to claim 48 are also pertinent to claim 84. In addition to the arguments above, the following arguments should also be taken into account.

Claim 84 claims:

A data processing system comprising a data storage device and a processor programmed to read data from, and write data to, said storage device, in which said storage device stores multiple operation records each storing data relating to one or more historical operation involving at least one entity,

each said entity being an identifiable thing within a business or other undertaking to which information resulting from a transaction, measurement or other such assignment can be related; and multiple entity records storing data indicating relationships between said entities,

wherein the entity records comprise a hierarchical structure, in which at least a first entity record relates to a specific entity, and a second to a more generic entity encompassing said specific entity, said entity records including link data linking said first and second entity records whereby to allow said processor to traverse said hierarchy,

said processor being arranged to generate output data by inputting instructions defining one or more selected entity dimensions across which said output data is to be distributed; and

if all required said operation records do not relate to entities of the dimension to which the operation records relate, the processor is programmed to determine, from said entity records, a hierarchically higher level entity dimension and to repeat said determination and, in the event that all required said operation records relate to said hierarchically higher level, to use said hierarchically higher entity instead of said selected entity in selecting said subset of operation records.

Page 36, line 3 to page 38, line 2 of the current application gives an example of how the system traverses the hierarchical levels in order to obtain as much useful data as possible when answering a query. This allows a user to obtain data from an organisation over a period of time, whether the organisation has undergone fundamental restructuring or not. Any restructuring could, in standard

relational databases, result in the loss of the relevant data required. Whereas in the system of the present invention, the data is stored in a manner such that data relevant before and after the organisational restructure is kept, and the hierarchical structure can be traversed by the system in order to extract relevant data from before and after any organisational restructuring.

It is admitted in the Office Action that Doktor does not explicitly disclose a "hierarchically higher level entity dimension wherein a determination is repeated and, in the event that all required said operation records relate to said hierarchically higher level, to use said hierarchically higher entity instead of said selected entity in selecting said subset of operation records." However, it is alleged in the Office Action that Eisenberg discloses this feature at col. 3, lines 20 – 32 and col. 24, lines 8 – 48. Applicants respectfully disagree. As discussed above in relation to claim 48, Eisenberg does not disclose the hierarchical arrangement of entities, but instead discloses a hierarchical arrangement of different sets of data (variants or versions). This is completely different than the structure as claimed and so claim 84 is patentable over the combination of Doktor and Eisenberg.

D. Claims 85 and 86

Claim 85, as amended, claims:

A data processing system comprising a data storage device and a processor programmed to read data from, and write data to, said storage device, in which said storage device stores:

a) multiple operation records each storing data relating to one or more historical operation involving at least one entity, each said operation record comprising data recording the operation, and data defining a date associated with the operation;

b) each said entity being an identifiable thing within a business or other undertaking to which information resulting from a transaction, measurement or other such assignment can be related, and each being represented by a single corresponding entity record; and

c) multiple entity relationship records storing data indicating relationships between said entities, and each said relationship being associated with a historical period of validity;

wherein the processor is programmed to extract data from a subset of said operation records and select said subset by the steps of:

inputting instructions defining one or more selected entities for which said output data relates; and

selecting said subset based on both the dates stored in said operation records and the historical periods of validity associated with the relationships of said selected entities.

Claim 86 claims:

A data processing system comprising a data storage device and a processor programmed to read data from, and write data to, said storage device, in which said storage device stores:

a) multiple operation records each storing data relating to one or more historical operation involving at least one entity, each said operation record comprising data recording the operation, and data defining a date associated with the operation;

b) each said entity being an identifiable thing within a business or other undertaking to which information resulting from a transaction, measurement or other such assignment can be related, and each being represented by a single corresponding entity record; and

c) multiple entity relationship records storing data indicating relationships between said entities, and each said relationship being associated with a historical period of validity;

wherein the processor is programmed to extract data from a subset of said operation records and select said subset to represent by the steps of:

inputting an analysis date;

for the selected entities, selecting the entity relationships which have associated historical periods of validity within which said analysis date lies; and

selecting said subset using those selected entity relationships.

Claims 85 and 86 have been amended to define the invention more clearly.

For at least the reasons discussed above in relation to claims 1 and 47 concerning the validity of the feature of one entity record per entity, Applicants respectfully assert that claims 85 and 86 are also novel and unobvious over the cited prior art.

E. Claim 87

Claim 87 claims:

A data processing system comprising a data storage device and a processor programmed to read data from, and write data to, said storage device, in which said storage device stores two types of data;

the first type of data being transaction data;

the second type of data consisting of metadata and data associated with at least one entity, said entity being an identifiable thing within a business or other undertaking to which information resulting from a transaction, measurement or other such assignment can be related;

both said metadata and said data associated with at least one entity having a historical period of validity associated with it.

The combination of storing both a data model (formed from metadata) and a data structure (formed from data associated with at least one entity), both being associated with a historical period of validity, is not disclosed or suggested by Doktor and/or Eisenberg. In particular at col. 13, lines 51 – 58 in Eisenberg, referenced by the Examiner, it is disclosed that a variant has a time stamp. This is merely a period of time when that particular version of the data structure was valid and does not relate to a historical period of validity for both metadata and data associated with at least one entity. Indeed, metadata is not mentioned anywhere in Eisenberg.

Further, referring to the second reference provided by the Examiner at col. 79, lines 4 – 8, Eisenberg merely discloses that: "For a particular search path, the constraint checker must check all variant times for every time interval for which there was an alteration. It need be only as fine grained as to all the variant start and end times for instances that exist which overlap with these time intervals." There is absolutely no reference in this paragraph to the storage of metadata and data associated with at

least one entity. Further, there is no reference to these two types of data having a historical period of validity. The only times referenced are the start and end times of a variant, or version of a data structure.

Finally, at col. 2, lines 19 – 21, Eisenberg merely discloses that the preferred approach to implementing versioning of databases is to provide direct versioning of entries in the DBMS, with the versioning managed by the DBMS to preserve the semantic validity of the data in the system. The present invention is not a versioning database as defined by Eisenberg and so this paragraph is not relevant to the invention as claimed in Claim 87. There is no reference in this paragraph to the storage of metadata and data associated with at least one entity. Further, there is no reference to these two types of data having a historical period of validity.

For at least the reasons set forth above, Applicants respectfully assert that claim 87 is novel and unobvious over Doktor and Eisenberg, alone or in combination.

V. Conclusion

In light of the above, it is respectfully requested that all outstanding rejections and objections be reconsidered and withdrawn. The Examiner is respectfully requested to telephone the undersigned if he can assist in any way in expediting issuance of a patent.

The authorization for the Commissioner to charge the fee for extension of time and any underpayment is provided in the Fee Transmittal accompanying this document.

Respectfully submitted,

Date: September 7, 2004

By: _____
Jeffrey R. Kurin
Reg. No. 41,132

Customer No. 23910
Fliesler Meyer LLP
Four Embarcadero Center, Fourth Floor
San Francisco, California 94111-4156
Telephone: 415/362-3800
Facsimile: 415/362-2928